Image Annotation using Moments and Multilayer Neural Networks

Mustapha Oujaoura
Laboratory of Information Processing and Telecommunications, Faculty of Science and Technology, Sultan Moulay Slimane University
PO Box. 523, Béni Mellal, Morocco

Brahim Minaoui
Laboratory of Information Processing and Telecommunications, Faculty of Science and Technology, Sultan Moulay Slimane University
PO Box. 523, Béni Mellal, Morocco

Mohammed Fakir
Laboratory of Information Processing and Telecommunications, Faculty of Science and Technology, Sultan Moulay Slimane University
PO Box. 523, Béni Mellal, Morocco

ABSTRACT
This document presents a system in order to annotate image content by using the region growing segmentation, as a method to separate different objects within an image, and the multilayer neural network to classify these objects and to find the appropriate keywords for them. In many applications, different kinds of moments have been used as features to classify the images and objects' shapes. The Hu moments, Legendre moments and Zernike moments are used, in this paper, as features to describe an image. The experiments are done through using ETH-80 database images.

General Terms
Neural Networks, Moments, Image Annotation.

Keywords
Image annotation, image segmentation, neural network, Zernike moments, Legendre moments, Hu moments.

1. INTRODUCTION
Recently, many reported works have been conducted in the context of image classification or categorisation which label images with a few high-level concepts that are defined in an application domain. Developing automatic methods to manage great quantity of digital information is increasingly important as soon as online resources are still an impulsory resource in our daily life. Yet, among these methods, automatic organizing and indexing of multimedia data remain an important challenge for researchers, especially the image annotation [1]. Annotated images play a very important role in information processing. They are useful for an image retrieval based on keywords and the image content management [2]. Since the quantity of information is unlimited, the manual annotation is not only boring but also not practical in many cases. That is to say, most images remain with inadequate annotations. Therefore, the automatic image content annotation becomes an interesting research [3] because it attempts to explore the visual characteristics of images and associate them with image contents and semantics.

Because many images are not adequately described or haven't been labeled yet, we suggest a system that tends to find the suitable annotation terms for each image so as to describe its individual components. The rest of this paper is moving to the section 2 which will present the annotation's system. Next, the section 3 is going to tackle the primordial task of an annotation's system as a segmentation problem. The section 4 provides the formulation for Hu moments, Legendre moments and Zernike moments as features extraction methods whereas the section 5 is about the classification by using neural network and the annotation's problem. Finally, the section 6 shows the results of the experimental annotation.

2. ANNOTATION SYSTEM
Due to the various sources of huge quantities of information in the media, the automatic image annotation is an effective technology to improve the image retrieval. The algorithms and systems used for image annotation are commonly divided into these tasks [4]:

- Segmentation,
- Feature extraction,
- Classification and Annotation.

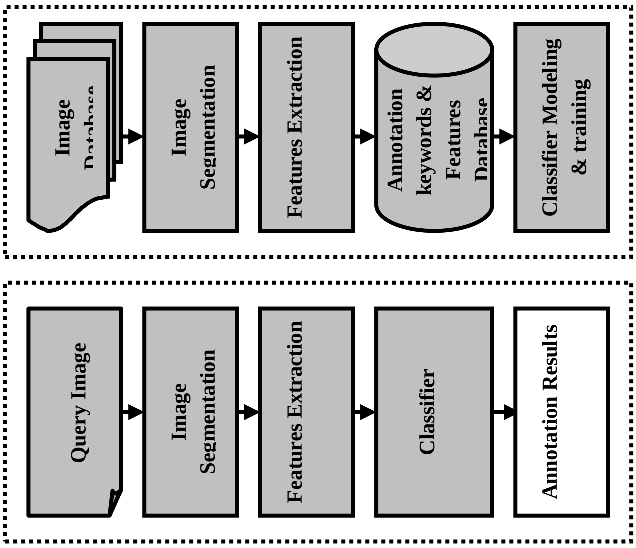


Fig 1: The annotation system.

The adopted annotation system, in this work, is shown in Fig.1. The system can be divided into two parts: the first one is about a reference's database of annotated images that have been already done by experts (manual annotation). This database is used for modelling and training the classifier. The second one can be considered as the image annotation, as the topic of this work. To achieve this goal, the query image is firstly segmented into regions that represent objects in the image. Secondly, the feature vectors of each region are computed and extracted from that query image. Those feature vectors are finally feed into the input of a neural network classifier in order to choose the appropriated annotation keywords.

3. IMAGE SEGMENTATION

Image segmentation is a method that localizes and extracts an object from an image or divides the image into several regions. It plays a very important role in many applications for image processing. It still remains a challenge for scientists and researchers despite the efforts and attempts that are being made to improve the segmentation techniques. Due to the improvement of computer processing capabilities, there are several possible segmentation techniques of an image: threshold, region growing, active contours, level sets, etc... [5] Among these methods, the region growing is well suited because of its simplicity and successful use, as a segmentation technique of digital images, several times. From a set of initial points, this method aggregates the pixels along two criteria: homogeneity and adjacency. This aggregation of pixels is controlled by a predicate (Boolean expression) that evaluates the need for evolution of segmentation. This process is applied to one or to more regions. For each region, it includes an initialization phase and an iterative one that changes the region until getting the object segmented.

The regions are iteratively grown by comparing all unallocated neighbouring pixels to the regions. The difference between a pixel's intensity's value and the region's average is used as a measure of similarity. It's a predicate that controls the evolution of segmentation. With the smallest difference measured this way, the pixel is allocated to the respective region. This process continues until all pixels are allocated to a region.

The objects are usually situated at the center of images. That's why the region growing segmentation is started from the corner of this image to isolate the objects in the center of it. The region growing segmentation algorithm used in this paper is presented in Fig.2. An example of image segmentation is also presented in Fig.3 as follows: (a) image during segmentation process and (b) the final result of image segmentation.

While an image is not entirely segmented;

1. Choose an unlabeled pixel p_k;

2. Set the region's average equally to the intensity of pixel p_k;

3. Consider unlabeled neighboring pixels p_{kj};

If (pixel's intensity - region's average) < threshold;

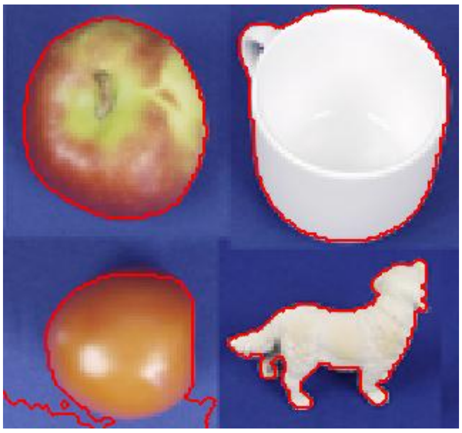
a. Put the pixel p_{kj} in the region labeled by k.

b. Update the region's average and go back to ③;

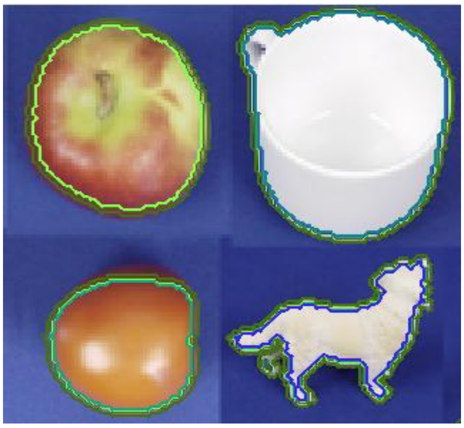
Else k = k + 1 and go back to ①.

Fig 2: The algorithm of region growing segmentation.

The feature vector, which is extracted from the entire image, loses local information. Therefore, it is necessary to segment an image into regions or objects of interest and use of local characteristics.



(a)



(b)

Fig 3: (a) Example of an image during segmentation process and (b) Example of the result of image segmentation.

Therefore, the objective of image segmentation is to divide the original image into several distinctive regions that correspond to the objects in a scene. After having distinctive regions, the vector's features are extracted from a region. These features can be considered as a representation of an object in the entire image.

4. FEATURES' EXTRACTION

When the input data to an algorithm is too large to be processed, it will be transformed into a reduced representation of features' set. Transforming the input data into the set of features is called features' extraction. The extraction task transforms rich contents of images into various features. In order to perform this task using this reduced representation instead of the full size input, we need to extract carefully these features [4]. It enhances not only the retrieval and annotation's accuracy, but also the annotation's speed. So a large image database can be organized according to the classification rule and therefore, the searching can be performed [6].

In this paper, we choose to use moments for feature's extraction from the segmented images. The use of moments for image analysis and pattern recognition was inspired by Hu [7]. The most common used moments are:

- Hu moments.
- Legendre moments.
- Zernike moments.

4.1 Hu moments

Hu moments are widely used in the image processing and pattern recognition. They are derived and calculated from geometric moments.

The two-dimensional geometric moments of order (p + q) of an image, that is represented by a real valued measurable function f(x, y) in the interval range [a1, a2] x [b1, b2], are defined as:

$$M_{pq} = \int_{a_1}^{a_2}\int_{b_1}^{b_2} x^p y^q f(x,y)\, dxdy \qquad (1)$$

Where p, q=0, 1, 2... ∞.

The monomial product x^p y^q is the basis function for this moment definition. A set of n moments consists of all Mpq for p+ q ≤ n.

The zero order moment, M00, of the function f(x, y)

$$M_{00} = \int_{a_1}^{a_2}\int_{b_1}^{b_2} f(x,y)\, dxdy \qquad (2)$$

represents the total of mass of the given image.

The two first order moments

$$M_{10} = \int_{a_1}^{a_2}\int_{b_1}^{b_2} x\ f(x,y)\, dxdy$$
$$M_{01} = \int_{a_1}^{a_2}\int_{b_1}^{b_2} y\ f(x,y)\, dxdy \qquad (3)$$

represent the centre of mass of the given image.

In terms of moment values, the coordinates of the centre of mass are

$$\bar{x} = \frac{M_{10}}{M_{00}} \quad , \quad \bar{y} = \frac{M_{01}}{M_{00}} \qquad (4)$$

The central moments of an image, that is represented by f(x, y), are defined as:

$$\alpha_{pq} = \int_{a_1}^{a_2}\int_{b_1}^{b_2} \left(x-\bar{x}\right)^p \left(y-\bar{y}\right)^q f(x,y)\, dxdy \qquad (5)$$

Where $\bar{x}$ *and* $\bar{y}$ are defined in (4).

The central moments α_{pq} defined in (5) are invariant under the translation of coordinates. They can be normalized to preserve the invariance by scaling. For $p+q=2,3,...$ The Normalized central moments of an image are given by:

$$\mu_{pq} = \frac{\alpha_{pq}}{\alpha_{00}^{\ \gamma}} \text{ , with } \gamma = \frac{p+q}{2}+1 \qquad (6)$$

Based on the theory of algebraic invariance, Hu [7] derived relative and absolute combinations of moments that are invariant to scale, position and orientation. Hu defined the following seven functions, computed from the normalized central moments through the order three, that are invariant to scale, translation and rotation changes:

$$\phi_1 = \mu_{20} + \mu_{02} \qquad (7)$$

$$\phi_2 = \left(\mu_{20} + \mu_{02}\right)^2 + 4\mu_{11}^2 \qquad (8)$$

$$\phi_3 = \left(\mu_{30} - 3\mu_{12}\right)^2 + \left(3\mu_{21} - \mu_{03}\right)^2 \qquad (9)$$

$$\phi_4 = \left(\mu_{30} + \mu_{12}\right)^2 + \left(\mu_{21} + \mu_{03}\right)^2 \qquad (10)$$

$$\begin{aligned}\phi_5 = &\left(\mu_{30} - 3\mu_{12}\right)\left(\mu_{30} + \mu_{12}\right)\times\\ &\left[\left(\mu_{30} + \mu_{12}\right)^2 - 3\left(\mu_{21} + \mu_{03}\right)^2\right]+\\ &\left(3\mu_{21} - \mu_{03}\right)\left(\mu_{21} + \mu_{03}\right)\times\\ &\left[3\left(\mu_{30} + \mu_{12}\right)^2 - \left(\mu_{21} + \mu_{03}\right)^2\right]\end{aligned} \qquad (11)$$

$$\begin{aligned}\phi_6 = &\left(\mu_{20} - \mu_{02}\right)\left[\left(\mu_{30} + \mu_{12}\right)^2 - \left(\mu_{21} + \mu_{03}\right)^2\right]\\ &+4\mu_{11}\left(\mu_{30} + \mu_{12}\right)\left(\mu_{21} + \mu_{03}\right)\end{aligned} \qquad (12)$$

$$\begin{aligned}\phi_7 = &\left(3\mu_{21} - \mu_{03}\right)\left(\mu_{30} + \mu_{12}\right)\times\\ &\left[\left(\mu_{30} + \mu_{12}\right)^2 - 3\left(\mu_{21} + \mu_{03}\right)^2\right]-\\ &\left(\mu_{30} - 3\mu_{12}\right)\left(\mu_{21} + \mu_{03}\right)\times\\ &\left[3\left(\mu_{30} + \mu_{12}\right)^2 - \left(\mu_{21} + \mu_{03}\right)^2\right]\end{aligned} \qquad (13)$$

The function ϕ_1 through ϕ_6 are invariant to rotation and reflection while ϕ_7 changes sign under reflection.

4.2 Legendre moments

Legendre moments, were first introduced by Teague [8]. Legendre moments are orthogonal moments. They were used in several patterns' recognition [9].

The $(p+q)^{th}$ order of Legendre moment, of an image with intensity function f(x, y) is defined on the square [-1,1] x [-1,1], is:

$$L_{pq} = \lambda_{pq} \int_{-1}^{+1}\int_{-1}^{+1} P_p(x)\ P_q(y)\ f(x,y)\, dxdy$$

(14)

Where $\lambda_{pq} = \frac{(2p+1)(2q+1)}{4}$, p,q=0,1,2,..,∞.

And $P_p(x)$ is the pth-order Legendre polynomial defined by:

$$P_p(x) = \sum_{k=0}^{p} a_{pk} x^k = \frac{1}{2^p\, p!} \frac{d^p}{dx^p}\left(x^2-1\right)^p$$

(15)

Or,

$$P_p(x) = \sum_{k=0}^{p} \left\{ \frac{(-1)^{\frac{p-k}{2}} x^k (p+k)!}{2^p k! \left(\frac{p-k}{2}\right)! \left(\frac{p+k}{2}\right)!} \right\}_{p-k=even} \quad (16)$$

The Legendre polynomials have the generating function:

$$\frac{1}{\sqrt{1-2rx+r^2}} = \sum_{p=0}^{\infty} r^p P_p(x) \quad , \quad r \;\pi\; 1 \qquad (17)$$

By deriving the two parts of generating function above, the recurrent formula of the Legendre polynomials can be acquired straightforwardly:

$$\frac{d}{dr}\left(\frac{1}{\sqrt{1-2rx+r^2}}\right) = \frac{d}{dr}\left(\sum_{p=0}^{\infty} r^p P_p(x)\right)$$

$$\frac{1}{\sqrt{1-2rx+r^2}} \times \frac{x-r}{1-2rx+r^2} = \sum_{p=0}^{\infty} p r^{p-1} P_p(x)$$

Then we have:

$$(x-r)\sum_{s=0}^{\infty} r^p P_p(x) = (1-2rx+r^2)\sum_{p=0}^{\infty} p r^{p-1} P_p(x)$$

And, the recurrent formula of the Legendre polynomials is:

$$\begin{cases} P_{p+1}(x) = \dfrac{2p+1}{p+1}\, x\, P_p(x) - \dfrac{p}{p+1} P_{p-1}(x) \\ P_1(x) = x \quad , \quad P_0(x) = 1 \end{cases} \qquad (18)$$

The Legendre polynomials are a complete orthogonal basis set on the interval [-1, 1]:

$$\int_{-1}^{+1} P_p(x) P_q(x) dx = \frac{2}{2p+1} \delta_{pq} \qquad (19)$$

Where $\delta_{pq} = \begin{cases} 1 & if \quad p = q \\ 0 & if \quad p \neq q \end{cases}$ is the Kronecker symbol.

The orthogonal property of Legendre polynomials implies no redundancy or overlapping of information between the moments with different orders. This property enables the contribution of each moment to be unique and independent from the information in an image [8].

Function LegendrePolynomial (x, p)
px=0 ;
for k=0 **to** p
    if mod (p-k,2)=0

$$c = \frac{(-1)^{\frac{p-k}{2}} x^k (p+k)!}{2^p k! \left(\frac{p-k}{2}\right)! \left(\frac{p+k}{2}\right)!} ;$$

        px=px+c;
    end if
end for
return px ;
Function LegendreMoments (p, q)
L=0;
for x=0 **to** (M-1)
    for y=0 **to** (N-1)

$$x_i = \frac{2x-(M-1)}{M-1}; \quad y_j = \frac{2y-(N-1)}{N-1};$$

        px= LegendrePolynomial (x_i, p) ;
        py= LegendrePolynomial (y_i, q) ;

$$L = L + f(x,y) * px * py ;$$

    end for
end for
return $\dfrac{L\,(2p+1)(2q+1)}{M \times N}$;

Fig 4: Pseudo code for computing Legendre moments.

To compute Legendre moments from a digital image, the integrals in (14) are replaced by summations and the coordinates of the image must be normalized into [-1, 1]. Therefore, the exact form of Legendre moments, for a discrete image of M x N pixels with intensity's function f(x, y), is:

$$L_{pq} = \lambda_{pq} \sum_{x=0}^{M-1}\sum_{y=0}^{N-1} P_p(x_i)\, P_q(y_j)\, f(x,y) \qquad (20)$$

Where $\lambda_{pq} = \dfrac{(2p+1)(2q+1)}{M \times N}$, x_i and y_j denote the normalized pixel coordinates in the range of [-1, +1], which are given by:

$$x_i = \frac{2x-(M-1)}{M-1} \quad and \quad y_j = \frac{2y-(N-1)}{N-1} \qquad (21)$$

The formula defined in (20) is obtained by replacing the integrals in (14) by summations and by normalizing the pixel coordinates of the image into the range of [-1, 1] using (21).

Fig. 4 shows the pseudo code for computing Legendre moments of order (p + q) by equation defined in (20) and by using direct method for calculating Legendre polynomials. In this work the recurrent formula is used for calculating Legendre polynomials in order to increase computation speed.

4.3 Zernike moments

Zernike moments, as a type of moment function, are the mapping of an image onto a set of complex Zernike polynomials. As these Zernike polynomials are orthogonal to each other, Zernike moments can represent the properties of an image with no redundancy or overlapping of information between the moments [10]. Due to these characteristics, Zernike moments have been used as features set in many applications [11].

The computation of Zernike moments from an input image consists of three steps: computation of radial polynomials, computation of Zernike polynomials, and computation of Zernike moments by projecting the image onto the Zernike polynomials.

The procedure of obtaining Zernike moments from an input image begins with the computation of radial polynomials. The real-valued radial polynomial is defined as:

$$R_{p,q}(r) = \sum_{s=0}^{(p-|q|)/2} \frac{(-1)^s (p-s)!\, r^{p-2s}}{s!\left(\frac{p+|q|}{2}-s\right)!\left(\frac{p-|q|}{2}-s\right)!} \quad (22)$$

With $R_{p,q}(r) = R_{p,-q}(r)$.

In (22), p and q are generally called respectively order and repetition. The order p is a non-negative integer, and the repetition q is an integer satisfying $p-|q| = (even)$ and $|q| \le p$. The radial polynomials satisfy the orthogonal properties for the same repetition q,

$$\int_0^{2\pi}\int_0^1 R_{p,q}(r,\theta) R_{p',q}(r,\theta) r dr d\theta = \frac{\delta_{pp'}}{2(n+1)} \quad (23)$$

Where $\delta_{pp'}$ is the Kronecker symbol defined above.

Using the radial polynomial, complex-valued 2-D Zernike polynomials, which are defined within a unit circle, are formed by:

$$V_{pq}(x,y) = V_{pq}(r\sin\theta, r\cos\theta) = R_{p,q}(r) e^{jq\theta} \quad (24)$$

Where, $j = \sqrt{-1}$, $|r| \le 1$ is the length of the vector from the origin to the pixel at (x, y), and θ is the angle between vector r and the x axis.

The Zernike polynomials are a complete set of complex-valued functions orthogonal on the unit disk $x^2 + y^2 \le 1$:

$$\iint_{x^2+y^2\le 1} [V_{mn}(x,y)]^* V_{pq}(x,y) dxdy = \frac{\pi\, \delta_{mp}\delta_{nq}}{m+1} \quad (25)$$

Or, in polar coordinates:

$$\int_0^{2\pi}\int_0^1 [V_{mn}(r,\theta)]^* V_{pq}(r,\theta) r dr d\theta = \frac{\pi\, \delta_{mp}\delta_{pq}}{m+1} \quad (26)$$

Where the Asterisk * denotes the conjugated complex.

The complex Zernike moments of order p with repetition q for an image function f(x, y) are finally defined as:

$$Z_{pq} = \frac{p+1}{\pi} \iint_{x^2+y^2\le 1} [V_{pq}(x,y)]^* f(x,y) dxdy \quad (27)$$

Or, in polar coordinates:

$$Z_{pq} = \frac{p+1}{\pi} \int_0^{2\pi}\int_0^1 [V_{pq}(r,\theta)]^* f(r,\theta) r\, dr\, d\theta \quad (28)$$

According to this definition, the procedure to compute Zernike moments can be seen as an inner product between the image's function and the Zernike polynomials.

To compute Zernike moments from a digital image, the integrals in (27) and in (28) are replaced by summations in addition to the coordinates of the image which must be normalized into [0, 1] by a mapping transform. The two commonly used cases of the transformations are shown in Fig. 5: (b) the image is over a unit circle and (c) the image is inside a unit circle. Based on the figure Fig. 5 (b), the pixels, that are located on the outside of the circle, are not involved in the computation of the Zernike moments. Accordingly, Zernike moments, which are computed by the mapping transformation, do not describe the properties of the outside of the unit circle in the image. This can be considered as a default while calculating Zernike moments. The discrete form of the Zernike moments of an image size M × N is expressed as follows:

$$\begin{aligned} Z_{pq} &= \frac{p+1}{\lambda} \sum_{x=0}^{M-1}\sum_{y=0}^{N-1} [V_{pq}(x,y)]^* f(x,y) \\ &= \frac{p+1}{\lambda} \sum_{x=0}^{M-1}\sum_{y=0}^{N-1} R_{pq}(r_{xy}) e^{-jq\theta_{xy}} f(x,y) \end{aligned} \quad (29)$$

Where $0 \le r_{xy} \le 1$ and λ is a normalization factor.

In the discrete implementation of Zernike moments, the normalization factor λ must be the number of pixels, that are located in the unit circle by the mapping transformation, corresponds to the area of a unit circle π in the continuous domain. The transformed phase θ_{xy} and the distance r_{xy} at the pixel of coordinates (x, y) are given by:

For Fig.1 (b):

$$\theta_{xy} = \tan^{-1}\left(\frac{(2y-(N-1))/(N-1)}{(2x-(M-1))/(M-1)}\right) \quad (30)$$

$$r_{xy} = \sqrt{\left(\frac{2x-(M-1)}{M-1}\right)^2 + \left(\frac{2y-(N-1)}{N-1}\right)^2} \quad (31)$$

And for Fig.1 (c):

$$\theta_{xy} = \tan^{-1}\left(\frac{2y-(N-1)}{2x-(M-1)}\right) \tag{32}$$

$$r_{xy} = \sqrt{\frac{(2x-(M-1))^2+(2y-(N-1))^2}{(M-1)^2+(N-1)^2}} \tag{33}$$

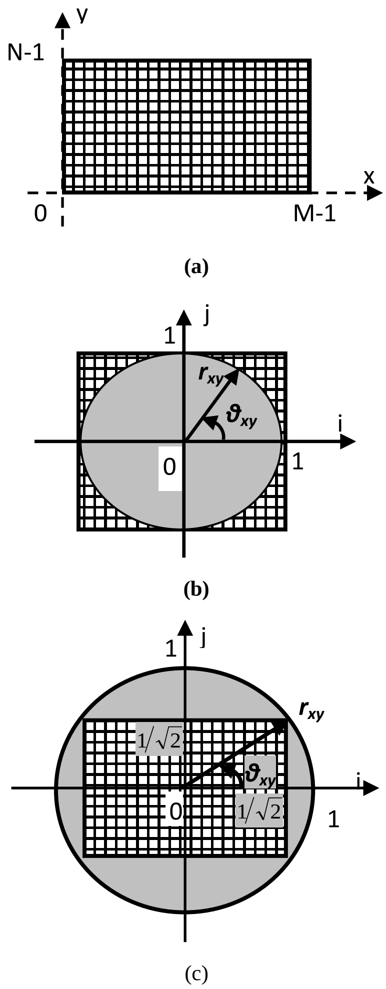


Fig 5: (a) Image M×N, (b) mapping of image over a unit circle and (c) mapping of image inside a unit circle.

Fig. 6 shows the pseudo code to compute Zernike moments of order p with repetition q by Eq. (30) and by using direct method for radial polynomials.

Function RadialPolinomial (r, p, q)
radial=0 ;
for s=0 **to** (p-q)/2

$$c = \frac{(-1)^s (p-s)!}{s!\left(\frac{p+|q|}{2}-s\right)!\left(\frac{p-|q|}{2}-s\right)!} \;;$$

radial=radial + c * r $^{p-2s}$;
end for
return radial ;
Function ZernikeMoments (p, q)
Zr=0; Zi=0;
Count=0;
for x=0 **to** (M-1)
for y=0 **to** (N-1)

$$r = \sqrt{\left(\frac{2x-(M-1)}{M-1}\right)^2 + \left(\frac{2y-(N-1)}{N-1}\right)^2} \;;$$

$$\theta = \tan^{-1}\left(\frac{(2y-(N-1))/(N-1)}{(2x-(M-1))/(M-1)}\right) ;$$

If $r \le 1$
Radial= RadialPolinomial (r, p, q) ;

$$Zr = Zr + f(x,y) * radial * \cos(q*\theta) ;$$

$$Zi = Zi + f(x,y) * radial * \sin(q*\theta) ;$$

Count=Count+1;
end if
end for
end for
return $\frac{(p+1)(Zr + i * Zi)}{Count}$;

Fig 6: Pseudo code for computing Zernike moments.

Most of the computation time of Zernike moments is because of computation of radial polynomials. Therefore, researchers have proposed faster methods that reduce the factorial terms by utilizing the recurrence relations on the radial polynomials.

Prata [12] proposed a recurrence's relation that uses radial polynomials of lower order than p as follows:

$$R_{pq}(r) = \frac{2rp}{p+q} R_{(p-1)(q-1)}(r) - \frac{p-q}{p+q} R_{(p-2)q}(r) \tag{34}$$

It is quite evident from the precedent equation that we can't compute all cases of p and q while computing the radial polynomials. It is not possible to use Prata's equation in cases where q =0 and p = q. Those cases can be obtained by other methods. The Direct method can be used in cases where q = 0, whereas the equation $R_{pp}(r) = r^p$ is used for p = q. The usage of direct method to compute radial polynomials in the case of q =0 will considerably increase the computation time especially when p is large.

Kintner [13] proposed another recurrence's relation that uses polynomials of a varying low-order p with a fixed repetition q to compute the radial polynomials as shown below:

$$R_{pq}(r) = \frac{(K_2 r^2 + K_3) R_{(p-2)q}(r) + K_4 R_{(p-4)q}(r)}{K_1} \tag{35}$$

Where the coefficients K1, K2, K3 and K4 are given by

$$K_1 = \frac{(p+q)(p-q)(p-2)}{2}$$

$$K_2 = 2p(p-1)(p-2)$$

$$K_3 = -q^2(p-1) - p(p-1)(p-2)$$

$$K_4 = \frac{-p(p+q-2)(p-q-2)}{2}$$

Like the equation in (35), Kintner's method can not be applied in cases where p = q and p − q = 2. For these two cases, in the normal approach, it is better to use the direct method although it takes too much time to compute. The following two relations are used to avoid the involvement of direct method. For p= q the equation $R_{pp}(r) = r^p$ is used, and for p − q = 2 the recurrent relation below is used:

$$R_{pq}(r) = pR_{pp}(r) - (p-1)R_{qq}(r) \tag{36}$$

This improved version of Kintner's method is denoted as modified Kintner's method.

Recently, Chong [15] presented the q-recursive method, which uses a relation of the radial polynomials of fixed order p and varying repetition of q. The relation of the radial polynomial is defined as

$$R_{pq}(r) = H_1 R_{p(q+4)}(r) + \left(H_2 + \frac{H_3}{r^2}\right) R_{p(q+2)}(r) \tag{37}$$

Where

$$H_1 = \frac{(q+4)(q+3)}{2} - (q+4)H_2 + \frac{H_3(p+q+6)(p-q-4)}{8}$$

$$H_2 = \frac{H_3(p+q+4)(p-q-2)}{4(q+3)} + (q+2)$$

$$H_3 = -\frac{4(q+2)(q+1)}{(p+q+2)(p-q)}$$

As the order p is fixed in (37), the individual order of Zernike moments can be calculated independently without referring to higher or lower order moments.

All these precedent methods focus only on the computation of Zernike radial polynomials and have some limitations if only a single Zernike moment is required because they use recurrence's relations. From the experiments in [14], the combined use of both the q-recursive method and modified Kintner's method takes the shortest time to compute a full set of Zernike moments followed by Kintner's method. Chong's method is much faster than other methods especially in computing Zernike moments with a fixed order. Therefore Chong's method is more effective in cases, where only selected orders of Zernike moments are used as feature vectors.

In this paper, we obtained Zernike moments using the hybrid method. The q-recursive method and modified Kintner's method are combined and used for calculation of the Zernike radial polynomials.

5. CLASSIFICATION AND ANNOTATION

The goal of pattern's classification is to allocate an object represented by a number of vector's features into one of a finite set of classes from the reference's database. In order to classify unknown patterns, a certain number of training samples, which are available for each class, are used to train the classifier [15]. The learning task is to compute a classifier or a model that approximate the mapping between the input-output examples and that labels correctly the training set with some levels of accuracy. This can be called the training or model generation stage. After being generated and trained, the model is able to classify an unknown instance, into one of the learned and labeled class in the training set. In other words, the classifier calculates the similarity of all the trained classes and assigns the unlabeled instance to the class with the highest measured similarity.

Consequently, image annotation can be approached by the generated and trained classifier to reduce the gap between low-level vector's feature and high-level concepts. The learned function can directly make the low-level features' set correspond to high-level conceptual classes.

Neural networks (or artificial neural networks) are learned by experience. they generalize previous experiences to new ones and can make decisions. A neural network can be considered as a black box non-parametric classifier [15]. Neural networks are therefore more flexible.

A multilayer network consists of an input layer including a set of input nodes, one or more hidden layers of nodes, and an output layer of nodes. Fig. 7 shows an example of a three layer network used in this paper, having input layer formed by m nodes, one hidden layer formed by 10 nodes, and output layer formed by n nodes. This neural network is trained to classify inputs according to target classes. The target data should consist of vectors of all zero values except for the one in element i, where i is the represented class. This type of neural network can approximate any function with the acceptable precision. The neural transfer function used, in this tree layer neural network, is a hyperbolic tangent sigmoid transfer function defined by:

$$tsig(x) = 2/(1+\exp(-2x))-1 \tag{38}$$

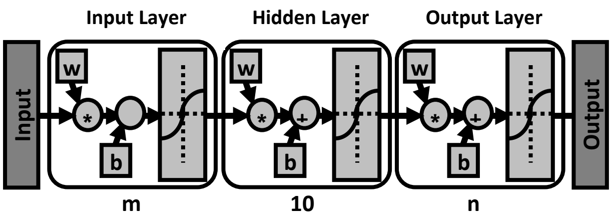


Fig 7: The three layer neural network.

For image annotation, low-level vector's features are calculated iteratively for each region in the image, either by using Hu moments, by Legendre moments or by Zernike moments. These vector's features are fed into the input layer of the neural network that is already trained, where each of the input neurons or nodes corresponds to each element of these features. And the output neurons of the neural network represent the class labels of images to be classified and annotated. Then each region is annotated by the corresponding label found by neural network classifier.

The input layer of the neural network has a variable number of input nodes. Concerning feature's extraction method, It has seven input nodes when adopting Hu moments, nine input nodes when adopting Zernike moments, and ten input nodes when using Legendre moment. However, the number of input nodes of the neural networks can be changed or increased when using Zernike and Legendre moments, as a feature's extraction method, in order to increase the accuracy of the annotation system.

6. EXPERIMENTS AND RESULTS

6.1 Experiments

In these used experiments, for each region that represent an object from the query image, the number of extracted inputs using Hu invariants feature's extraction method are 7 (hu1, hu2, hu3, hu4, hu5, hu6, hu7) while the number of extracted inputs using the order 4 of Zernike moments are 9 (Z00, Z11, Z20, Z22, Z31, Z33, Z40, Z42, Z44) and the number of extracted inputs using the order 3 of Legendre moments are 10 (L00, L01, L02, L03, L10, L11, L12, L20, L21, L30). These inputs are presented and fed to the neural network classifier to test and to match the feature values and the reference's database. Then, the appropriate keywords are selected and used for annotation of the query image. The fig. 8 shows some examples of the image objects from ETH-80 image's database used in our experiments. The experiments are made based on eight classes of objects (Apple, Car, Cow, Cup, Dog, Horse, Pears, and Tomato) as shown in Fig. 8.

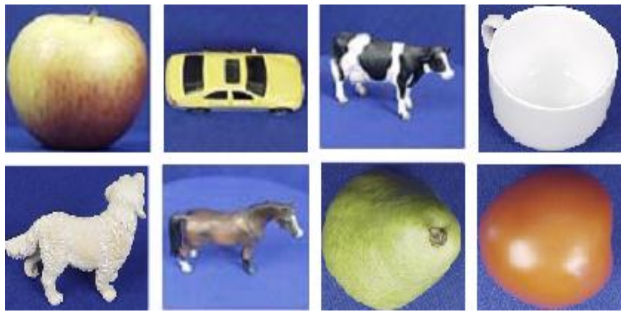


Fig 8: Some examples of objects from ETH-80 database.

The accuracy of image annotation is evaluated by the F1 measure which is an integrated value of the precision rate and recall rate. The precision rate P is the number of correct results divided by the number of all returned results and the recall rate R is the number of correct results divided by the number of results that should have been returned. The F1 measure can be interpreted as a weighted average of the precision and recall rates, where an F1 score reaches its best value at 1 and its worst value at 0. The consequent F1 measure for evaluating the accuracy of image annotation is [16]:

$$F_1 = \frac{2PR}{P+R} \tag{39}$$

All the experiments are conducted using.

All the tests and the experiments are performed using ETH-80 database containing a set of 8 different object images [17]. The proposed system has been implemented and tested on a core 2 Duo personnel computer using a Matlab software.

6.2 Results

The annotation's rates of Hu , Zernike and Legendre Moments as feature's extraction method are given for each objects in Table 1.

Table 1. Objects Annotation Scores of Hu, Zernike and Legendre Moments using Neural Network classifier

Object	Hu	Zernike	Legendre	Average
Apple	79,54%	79,54%	92,13%	83,74%
Car	68,53%	83,49%	62,18%	71,40%
Cow	25,14%	41,17%	44,39%	36,90%
Cup	76,57%	58,91%	73,20%	69,56%
Dog	51,57%	70,78%	84,61%	68,99%
Horse	31,83%	53,33%	50,28%	45,15%
Pears	79,99%	69,54%	70,78%	73,44%
Tomato	76,06%	66,49%	76,52%	73,02%
Average	61,15%	65,41%	69,26%	65,27%

The general annotation's rates and error rates of Hu moments, Zernike moments and Legendre Moments, as feature's extraction method, are given in Table 2. The experimental results shows that the annotation's rate of the neural network classifier based on Legendre moments is higher than the annotation's rate of Zernike and Hu moments.

Table 2 Annotation Score and Error Rate of Hu, Zernike and Legendre Moments using Neural Network classifier

Features Extraction method	Annotation Score	Error Rate
Hu moments	61,15%	38,85%
Zernike moments	65,41%	34,59%
Legendre moments	69,26%	30,74%
Average	65,27%	34,73%

The Fig. 9 presents an example of annotation results obtained using the presented system and Zernike moment as feature's extraction method.

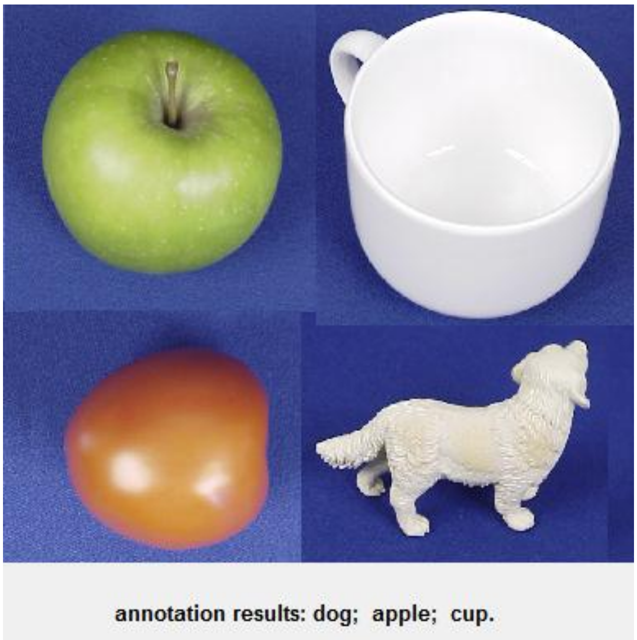


Fig 9: example of annotation results using Zernike moment as feature's extraction method.

The results are also affected by the accuracy of the image segmentation method. In most cases, it is very difficult to have an automatic and ideal segmentation. This problem decreases the annotation's rates. Therefore, any annotation attempt should consider the image segmentation as an important step, not only for automatic image annotation system, but also for all the systems which require an image segmentation. The Legendre moments and Zernike moments are very expensive regarding to the processing and computation time. So, any use of them in real time for an online image annotation system will be difficult and impracticable.

7. CONCLUSION

In this paper, we presented an image annotation system using region growing as image segmentation algorithm. For this image annotation system, we discussed the effect of using Hu moments, Legendre moments and Zernike moments as feature extraction methods. The neural network is chosen and trained to classify and annotate the input image by the suited keywords which are selected from the image's reference database. The performance of each feature extraction method has been experimentally analyzed. The successful experimental results proved that the proposed image annotation system gives good and modest results for some images that are well and properly segmented. However, the image segmentation remains a challenge that needs more care to increase the precision and the accuracy of the image annotation system. Also, the gap between the low-level features and the semantic content of an image must be reduced and be considered for more accuracy of any image annotation system. This system can be also improved by incorporating other feature descriptors such as texture, color,… and other classifiers can be included and compared to the neural networks.

8. REFERENCES

[1] N. Vasconcelos and M. Kunt, Content-Based Retrieval from Image Databases: Current Solutions and Future Directions, Proc. Int'l Conf. Image Processing, 2001.

[2] A. Smeulders, M. Worring, S. Santini, A. Gupta, and R. Jain, Content-Based Image Retrieval: The End of the Early Years, IEEE Trans. Pattern. Analysis and Machine Intelligence, vol. 22, no. 12, pp. 1349-1380, Dec. 2000.

[3] Lei Ye, Philip Ogunbona and Jianqiang Wang, Image Content Annotation Based on Visual Features, Proceedings of the Eighth IEEE International Symposium on Multimedia (ISM'06), IEEE computer society, San Diego, USA, 11-13 December 2006.

[4] Ryszard S. Chora´s, Image Feature Extraction Techniques and Their Applications for CBIR and Biometrics Systems, International Journal Of Biology And Biomedical Engineering, Issue 1, Vol. 1, pp. 6-16, 2007.

[5] Frank Y. Shih, Shouxian Cheng, Automatic seeded region growing for color image segmentation, Image and Vision Computing 23, pp. 877–886, 2005.

[6] Zijun Yang and C.-C. Jay Kuo, Survey on Image Content Analysis, Indexing, and Retrieval Techniques and Status Report of MPEG-7, Tamkang Journal of Science and Engineering, Vol. 2, No. 3, pp. 101-118, 1999.

[7] Gabriel Taubin , David B. Cooper, Object Recognition Based on Moment (or Algebraic) Invariants, Geometric Invariance in Computer Vision, J.L. Mundy and A.Zisserman, eds., MIT Press, pp. 375-397, 1992.

[8] Rao Ch. Srinivasa, Kumar S. Srinivas, Mohan B. Chandra, Content Based Image Retrieval Using Exact Legendre Moments and Support Vector Machine, The International Journal of Multimedia & its Applications (IJMA), Vol. 2, No 2, pp. 69-79, May 2010.

[9] Chee-Way Chonga, P. Raveendranb and R. Mukundan, Translation and scale invariants of Legendre moments, Pattern Recognition 37, pp. 119 – 129, 2004.

[10] Sun-Kyoo Hwang, Whoi-Yul Kim, Anovel approach to the fast computation of Zernike moments, Pattern Recognition 39, pp. 2065 – 2076, 2006.

[11] Sinan Tumen, M. Emre Acer and T. Metin Sezgin, Feature Extraction and Classifier Combination for

Image-based Sketch Recognition, Eurographics Symposium on Sketch-Based Interfaces and Modeling, pp. 1–8, 2010.

[12] A. Prata, W.V.T. Rusche, Algorithm for computation of Zernike polynomials expansion coefficients, Appl. Opt. 28, pp. 749–754, 1989.

[13] E.C. Kintner, On the mathematical properties of the Zernike polynomials, Opt. Acta. 23 (8) pp. 679–680, 1976.

[14] C.W. Chong, P. Raveendran, R. Mukundan, A comparative analysis of algorithms for fast computation of Zernike moments, Pattern Recognition 36 (3) , pp. 731–742, 2003.

[15] Haykin S. Neural networks: a comprehensive foundation, 2nd Edition. Prentice Hall, New Jersey. 1999.

[16] Yue Cao, Xiabi Liu, Jie Bing and Li Song, Using Neural Network to Combine Measures of Word Semantic Similarity for Image Annotation, IEEE International Conference on Information and Automation (ICIA), pp. 833 – 837, 2011.

[17] ETH-80 database image. [Online]. Available: http://www.d2.mpi-inf.mpg.de/Datasets/ETH80